



# Software Testing with the approach of Blackbox Testing on the Academic Information System

Supriyono

Department of Informatics, Faculty of Science and Technology  
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim, Malang, Indonesia

[priyono@ti.uin-malang.ac.id](mailto:priyono@ti.uin-malang.ac.id)

## Abstract

*SIAKAD is a website based information system assessment application that holds academic data of an educational institution. SIAKAD SDIT Robbani was created to help the assessment process of students, both in the form of public and religious subjects. With this investigation is expected to facilitate the assessment process at the school Rabbani. For that, testing is required to know whether the system can run as expected or not. Software Testing was conducted using black-box testing. The software testing method uses a Boundary Value Analysis (BVA) because the assessment has the minimum and maximum constraints of an inputted value. The test results show the scoring system has not been able to fulfil expectations because there is no limit value so the random value will remain stored by the system. This can result in incorrect information generated by the SIAKAD.*

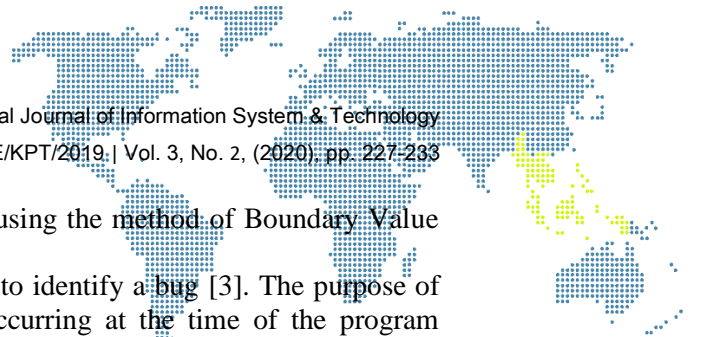
**Keywords:** *Software Testing, Boundary Value Analysis, Black-box Testing, Information System.*

## 1. Introduction

The academic information system is a website-shaped information system that holds the academic data of an educational institution. SIAKAD SDIT Rabbani is one of the sections on the implementation of the Web-based academic information system. SIAKAD SDIT Robbani was made to assist the student assessment process, and there are several modules such as semester assessment, replay, and daily value of both general and religious studies. This SIAKAD system also provides a bank problem, print rapport and a report of subjects that students follow. With this application is expected, the assessment process in this basic is able to run well and more organized. The assessment of software is required with the goal of a better outcome. In the academic world, learning applications also use software applications such as hybrid learning mechanisms [1].

In the assessment section, there are three sections, semester assessments, replay assessments, and daily assessments. In this assessment, there are several values that must be entered such as the value of subjects, the value of worship, the value of Muroja'ah, the value of o, the value of the Qur'an, the value of personality, and the value of students attendance. This system is only used for teachers in SDIT Rabbani by entering a username and password for the access rights to manipulate student value data. The software that was developed required the optimization techniques such as examples in research related to green open spaces by optimizing the method with the expectation of good results [2].

Once the application is created, then the testing phase is next. Testing the software is done to determine whether the information system made can run in accordance with expectations or not. In conducting quality testing of SIAKAD SDIT Rabbani, it is done using the black-box testing method. Black Box testing is a software testing technique that works on the functional specifications of the software. One of the black box testing methods is the boundary value analysis (BVA). This method is focused on the input process where each value inputted there outing class is a minimum and maximum limit. The BVA method is very suitable to be applied in the scoring system, so the author is



interested in choosing the title "Black Box Testing using the method of Boundary Value Analysis in SIAKAD SDIT Rabbani".

Software testing is used to evaluate the software to identify a bug [3]. The purpose of this software testing is to detect possible errors occurring at the time of the program execution as well as the estimated performance of the system during tests [4]. This black box test assumes software as a "black box" unknown to its internal structure [5]. Black Box testing is one of the methods of testing software that discusses its functionality. The results of black-box testing influence human resource management (HRM) as it affects the satisfaction and performance of the system in making good decisions on an organization or institution [6]. The software implementation needs a mature system engineering and more specific planning of software [7]. The software in the process of testing required techniques and optimizations are quite intensive so that the results obtained satisfactorily [8].

## 2. Research Methodology

### 2.1. Black Box Testing

A test case is a description of the specific interaction that the tester will have to test the single behaviour of the software [9]. Black box testing considers the value of inputs on a study [10] and ignores the internal mechanisms of a system [11]. Currently, there are ten techniques to perform black-box testing methods [12] namely:

- a) Boundary Value Analysis (BVA)
- b) Cause-Effect Graphing (CEG)
- c) Decision Tables (DT)
- d) Equivalence Partitioning (EP)
- e) Orthogonal Array Testing (OAT)
- f) Random Testing (RT)
- g) Specification-Based Mutation Testing (SBMT)
- h) State-Transition Diagram Testing (STT)
- i) Syntax Testing (ST)
- j) Worst Case Testing (WCT)

The black box testing methodology consists of three main steps [13]. The three main steps are described as follows:

- a) Test case definitions, either using a graphical user interface or a text editor. A collection of test cases is stored in a test case library and can be reused in multiple testing processes.
- b) Execute test case, perform the result evaluation algorithm, and record the test results in the database.
- c) Test result management and testing report generation, which enhances the usability of Black-Box Testing

### 2.2. Boundary Value Analysis

The Boundary value analysis is one of the black box testing methods. BVA is also known as limit testing. This method performs testing against the top edge and the lower edge [14] of the value that is inputted in the system. There are several principles underlying the BVA, namely:

- a) If there are limited  $x$  and  $y$  limitations, Test  $(x-1)$ ,  $x$ ,  $(x + 1)$ , and  $(y-1)$ ,  $y$ ,  $(y + 1)$
- b) If the input requires a number of  $n$  values, then test  $(n-1)$ ,  $n$ , and  $(n + 1)$
- c) Apply the previous two rules on the output condition (for maximum and minimum values)

The BVA implementation aims to keep one variable at the normal or average value. So for any input value outside the normal number, it will give an error message. SIAKAD is

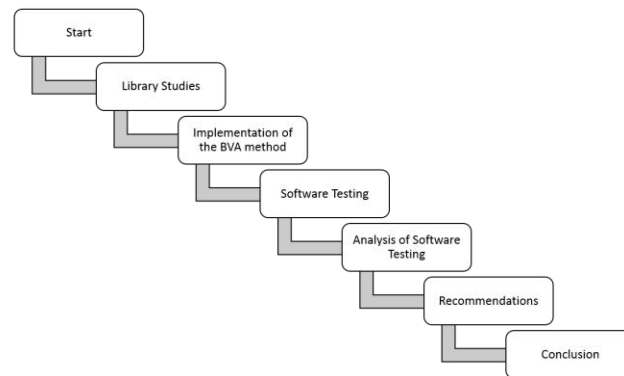
an abbreviation of an academic information system designed to handle the process of managing academic data, decision making to reporting in academia environment. The author performs a black box test with a Boundary Value Analysis (BVA) method on Siakad Rabbani. More specifically, the author will test the assessment menu, which includes daily assessments, replay, and semester.

The research mechanism is all the necessary processes in the planning and implementation of research. The research mechanisms to be implemented include several phases, as follows:

Search for reference journals and supporting journals regarding black-box testing and boundary value analysis (BVA).

- a) Implementing BVA on url:(<http://alhamra.siakad.cendana2000.biz/>)
- b) Conduct BVA testing on the website
- c) Analysis of results from BVA testing
- d) Draw conclusions from BVA testing on the website

The research mechanism can be seen in Figure 1.

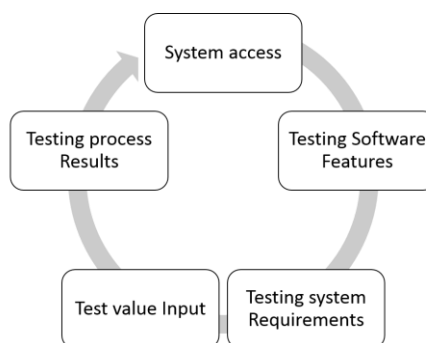


**Figure 1. Research Methods**

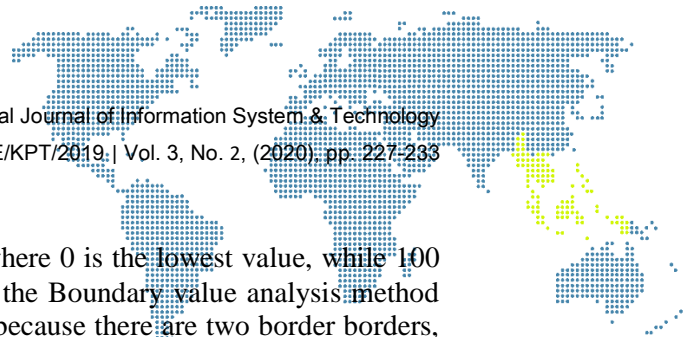
The phases of the following study include

- a. Library Studies
- b. Implementation of the BVA method
- c. Software Testing
- d. Analysis of Software Testing
- e. Recommendations
- f. Conclusion

Figure 2 shows the flow from the BVA test stage in the SIAKAD Assessment section.



**Figure 2. Testing Flow**



### 3. Results and Discussion

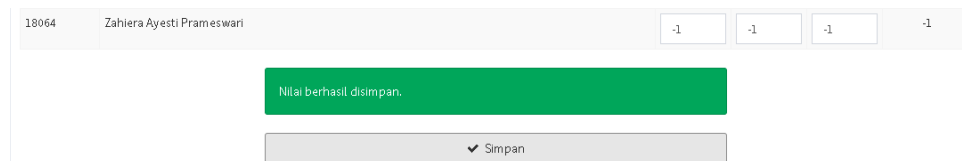
The following tests use a value range of 0-100 where 0 is the lowest value, while 100 is the highest value a student can reach. Testing of the Boundary value analysis method uses formulas  $(x-1)$ ,  $x$ ,  $(x+1)$  and  $(y-1)$ ,  $y$ ,  $(y+1)$  because there are two border borders, values 0 and 100. On the web implementation of SIAKAD SD IT Robbani shows in Figure three if the input value is 0 ( $x$ ) can be stored on the system. This indicates the test results in accordance with the expectations.



The screenshot shows a user interface with a header containing the ID '18064' and the name 'Zahiera Ayesti Prameswari'. Below the header, there are four input fields, each containing the value '0'. A green message box displays 'Nilai berhasil disimpan.' (Value successfully saved). At the bottom, there is a button labeled 'Simpan' with a checkmark icon.

**Figure 3. Testing Value  $X = 0$  Can Be Saved**

Further testing software with value  $x-1$  and the input value means negative  $(-1)$ . The test results appear, as shown in 4 below, which should be no negative value. This signifies the test result not yet in accordance with expectations.



The screenshot shows a user interface with a header containing the ID '18064' and the name 'Zahiera Ayesti Prameswari'. Below the header, there are four input fields, each containing the value '-1'. A green message box displays 'Nilai berhasil disimpan.' (Value successfully saved). At the bottom, there is a button labeled 'Simpan' with a checkmark icon.

**Figure 4. Testing Value  $X = 1$  Can Be Saved**

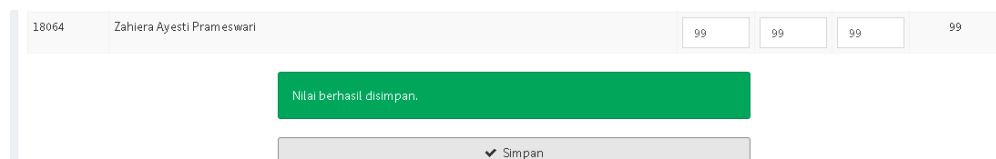
At the test shown as in figure 5 where  $x+1$  means the input value is  $0 < x < 100$ . In this test, the value  $x = 1$  and can be stored into the system database. It demonstrates conformity with expectations.



The screenshot shows a user interface with a header containing the ID '18064' and the name 'Zahiera Ayesti Prameswari'. Below the header, there are four input fields, each containing the value '1'. A green message box displays 'Nilai berhasil disimpan.' (Value successfully saved). At the bottom, there is a button labeled 'Simpan' with a checkmark icon.

**Figure 5. Testing value  $x + 1$  can be saved**

The software testing shown in Figure 6 is a limitation of the maximum value where the value of  $y = 100$  so that the test is carried out with a  $Y-1$  value. As shown in Figure 6, the value data can be stored so that it matches the expectation.



The screenshot shows a user interface with a header containing the ID '18064' and the name 'Zahiera Ayesti Prameswari'. Below the header, there are four input fields, each containing the value '99'. A green message box displays 'Nilai berhasil disimpan.' (Value successfully saved). At the bottom, there is a button labeled 'Simpan' with a checkmark icon.

**Figure 6. Testing value  $y-1$  can be saved**

If the value is entered  $y = 100$ , the data can also be saved into the database. The trial is shown in Figure 7.

18064	Zahiera Ayesti Prameswari	100	100	100	100
-------	---------------------------	-----	-----	-----	-----

Nilai berhasil disimpan.

✓ Simpan

**Figure 7. The n-1 value can be saved (daily value)**

However, if inserted  $y > 100$ , it should not be stored in the database because the value 100 is the maximum input value limit. Because of the assessment system, there is no limit of Min and Max, so the value of 1000 can still be saved as in Figure 8.

18062	Uwais Al-Qorni Abdul Rahman		0
18063	Vania Azkiya Rai'sa Noor	1000	1000

Nilai berhasil disimpan.

✓ Simpan

**Figure 8. The values = 1000 can be saved (daily value)**

The prayer assessment Menu is not given a range of grades, which is crucial because if the origin of the input, then it will be influential in the resulting output. In Figure 9, the author fills all the values with 1000 and remains able to be saved. Similar things also occur in other parts of the assessment, such as Muroja'ah and extracurricular.

5. Sujud	1000	5. Duduk Iftirosy	1000
6. Duduk Iftirosy	1000	6. Tahiyat Awal	1000
7. Tahiyat Awal	1000	7. Dzikir	1000
8. Tahiyat Akhir	1000	8. Doa Untuk Orang Tua	1000
9. Salam	1000	9. Doa Kebaikan Dunia Dan Akhirat	1000

Nilai berhasil disimpan

✓ Simpan

**Figure 9. The Value = 1000 can be stored (prayer assessment)**

In Figure 10, authors input the number of student attendance each month and not given maximum value, i.e. there are at least 31 days.

Mata Pelajaran	Ibadah	Muroja'ah	Ekstrakurikuler	Al Quran	Kepribadian	Kehadiran
Januari						Alpha 1000
Februari						Sakit 1000
Maret						Ijin 1000
April						
Mei						
Juni						
Juli						
Agustus						

Nilai berhasil disimpan

✓ Simpan

**Figure 10. Number of inputs 1000 can be saved (presence)**

Table 1 is an overview of the black box test results using the boundary value analysis method of the assessment system. With no limitations on the value to insert, it cannot be known the values of  $(x-1)$ ,  $x$ , nor  $(x + 1)$  so that the random value that is included will be stored into the database.

**Table 1. Boundary Value Analysis Test Data.**

Information	Input	Results	Results Hope	Output	Conclusion	Figure
Range Value	$x = 0$	$x-1$	The value of $< 0$ cannot be saved, and the system gives an error message	The value of $< 0$ can be stored by the system because the system has no value restrictions.	Failed	Figure 4
		$x$	Value $= 0$ can be saved	Value $= 0$ can be stored in the system.	Successful	Figure 3
		$x+1$	Value $> 0$ can be saved	Value 1 can be stored in the system.	Successful	Figure 5
	$y = 100$	$y-1$	The value of $< 100$ can be saved	value 99 can be saved in system managed	Successful	Figure 6
		$y$	value $= 100$ can be saved	value $= 100$ can be saved in system managed	Successful	Figure 7
		$y+1$	Value $> 100$ can not be saved, and the system gives an error message.	Value $= 1000$ can be stored by the system because it has no value limitation	Failed	Figure 8

#### 4. Conclusion

Based on the results of the tests described in the previous chapter using the Boundary Value Analysis (BVA) method in black box testing it can be concluded that testing applications Siakad SDIT Rabbani has not been able to fulfil expectations because there are still some shortcomings. This is because the scoring system does not give students the minimum and maximum value that a teacher can provide to the student. The same is also the case for the student attendance per month where any value of the presence that is inputted will still be preserved. Therefore, it can result in improper output where information is not delivered correctly.

#### References

- [1] Sutiah and Supriyono, "Improvement of e-learning based on hybrid learning methods at the university in the era of industrial revolution 4.0," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 6 Special Issue, pp. 2137–2142, 2020.
- [2] I. B. Santoso, Supriyono, C. Crysdiyan, and K. F. H. Holle, "Optimization of naïve bayes classifier to classify green open space object based on google earth image," *2018 Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2018*, pp. 465–469, 2018.
- [3] P. M. Jacob and M. Prasanna, "A Comparative analysis on Black box testing strategies," *Proc. - 2016 Int. Conf. Inf. Sci. ICIS 2016*, pp. 1–6, 2017.
- [4] G. Betta, D. Capriglione, A. Pietrosanto, and P. Sommella, "A statistical approach for improving the performance of a testing methodology for measurement software," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 6, pp. 1118–1126, 2008.
- [5] T. Y. Chen and P. L. Poon, "Experience With Teaching Black-Box Testing in a Computer Science/Software Engineering Curriculum," *IEEE Trans. Educ.*, vol. 47, no. 1, pp. 42–50, 2004.



- [6] F. Edgar and A. Geare, "Inside the 'black box' and 'HRM,'" *Int. J. Manpow.*, vol. 30, no. 3, pp. 220–236, 2009.
- [7] S. Supriyono and S. Sutiah, "Improvement of Project Management Using Accelerated SAP Method in the Odoo ERP," pp. 1–9, 2020.
- [8] S. Sutiah and S. Supriyono, "Fuzzy topsis optimization on expert systems for core competency detection and Islamic religious education student learning achievement at Perguruan Tinggi Keagamaan Islam Negeri (PTKIN)," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 732, no. 1, 2020.
- [9] H. Wu, "An effective equivalence partitioning method to design the test case of the WEB application," *2012 Int. Conf. Syst. Informatics, ICSAI 2012*, no. Icsai, pp. 2524–2527, 2012.
- [10] A. P. Estrada-Vargas, E. López-Mellado, and J. J. Lesage, "A Black-Box Identification Method for Automated Discrete-Event Systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 3, pp. 1321–1336, 2017.
- [11] M. A. Khan and M. Sadiq, "Analysis of black box software testing techniques: A case study," *Proc. 2011 Int. Conf. Work. Curr. Trends Inf. Technol. CTIT'11*, no. i, pp. 1–5, 2011.
- [12] T. Murnane, K. Reed, and R. Hall, "On the learnability of two representations of equivalence partitioning and boundary value analysis," *Proc. Aust. Softw. Eng. Conf. ASWEC*, pp. 274–283, 2007.
- [13] D. Marijan, N. Teslic, M. Temerinac, and V. Pekovic, "On the effectiveness of the system validation based on the black box testing methodology," *2009 IEEE Circuits Syst. Int. Conf. Test. Diagnosis, ICTD'09*, 2009.
- [14] W. Feng, "A generalization of boundary value analysis for input parameters with functional dependency," *Proc. - 9th IEEE/ACIS Int. Conf. Comput. Inf. Sci. ICIS 2010*, pp. 776–781, 2010.

## Authors



### **Supriyono, S.Kom., M.Kom**

Education that has been taken in the Department of Computer science at Universitas Brawijaya Malang and the Magister Program of Informatics Engineering of Institut Teknologi 10 Nopember Surabaya in the field of software engineering. Currently, the Lecturer majoring in the Department of informatics at Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang. Current research is related to system needs engineering, software quality, software management, and software development using several methods applied as well as in general, the research conducted in the centration in the field of software engineering.